

SAM2MOT: A Novel Paradigm of Multi-Object Tracking by Segmentation

Junjie Jiang^{1*}, Zelin Wang^{1*}, Manqi Zhao¹, Yin Li¹, Dongsheng Jiang^{1†}

¹Huawei Cloud

{jiangjunjie24, wangzelin14, jiangdongsheng1, zhaomanqi, liyin9}@huawei.com

Abstract

Inspired by Segment Anything 2, which generalizes segmentation from images to videos, we propose SAM2MOT—a novel segmentation-driven paradigm for multi-object tracking that breaks away from the conventional detection-association framework. In contrast to previous approaches that treat segmentation as auxiliary information, SAM2MOT places it at the heart of the tracking process, systematically tackling challenges like false positives and occlusions. Its effectiveness has been thoroughly validated on major MOT benchmarks. Furthermore, SAM2MOT integrates pre-trained detector, pre-trained segmentor with tracking logic into a zero-shot MOT system that requires no fine-tuning. This significantly reduces dependence on labeled data and paves the way for transitioning MOT research from task-specific solutions to general-purpose systems. Experiments on DanceTrack, UAVDT, and BDD100K show state-of-the-art results. Notably, SAM2MOT outperforms existing methods on DanceTrack by +2.1 HOTA and +4.5 IDF1, highlighting its effectiveness in MOT.

Code — <https://github.com/TripleJoy/SAM2MOT>

Introduction

Multi-object tracking (MOT) has long relied on the dominant detection-association paradigm (Bewley et al. 2016; Bergmann et al. 2019; Wu et al. 2021; Zhang et al. 2022; Cao et al. 2023). While this architecture is intuitive, it reveals structural vulnerabilities on two fronts. First, in crowded or occluded scenes, frequent interactions between objects lead to unstable detections, making association highly error-prone. Second, it heavily depends on the detector’s performance—when detections are ambiguous, tracking often collapses. This raises a fundamental question: should we continue to follow a paradigm that has already exposed its inherent limitations?

To address the shortcomings of the detection–association paradigm, recent studies propose end-to-end approaches (Sun et al. 2020; Wang et al. 2021; Meinhardt et al. 2022; Zeng et al. 2022) that jointly model detection and association or directly learn identity continuity. Though

*These authors contributed equally.

†Corresponding author.

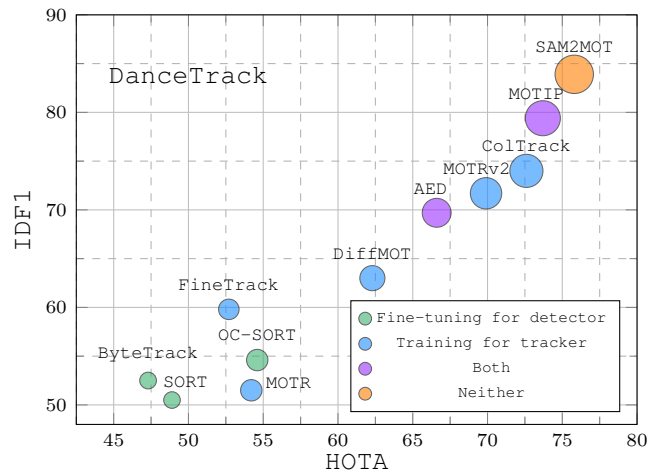


Figure 1: IDF1-HOTA-AssA comparisons of different trackers on the test set of DanceTrack, where the horizontal axis represents HOTA, the vertical axis represents IDF1, and the circle radius indicates AssA. This comparison highlights our method’s superior capability in associating objects across frames, surpassing all previous trackers.

conceptually promising, these methods face a major limitation—their dependence on large-scale, high-quality tracking data. Unfortunately, since annotating identities across frames is costly and labor-intensive, such data is scarce, leaving end-to-end models prone to unreliable associations and poor generalization under limited supervision.

These challenges compel us to fundamentally rethink the design of multi-object tracking architectures—toward approaches that go beyond detection reliance and generalize effectively in data-scarce scenarios.

Segmentation, though offering pixel-level precision and strong spatial awareness, has long remained peripheral in the MOT community. We argue that the root cause lies in existing methods largely treating segmentation as auxiliary information, lacking systematic modeling and dedicated architectural design. As a result, segmentation-based approaches consistently underperform on major benchmarks, further eroding confidence in their effectiveness. Inspired by Segment Anything 2 (SAM2), we revisit segmentation’s role in

video understanding and explore its potential as a core tracking mechanism to overcome current structural limitations.

Building on this insight, we introduce SAM2MOT, a segmentation-driven paradigm for multi-object tracking. It incorporates a cross-object interaction mechanism to systematically model and reduce identity ambiguity under occlusion. Furthermore, we design a unified framework that integrates a pre-trained detector and segmentor with tracking logic, enabling robust zero-shot identity tracking without scene-specific fine-tuning.

To the best of our knowledge, SAM2MOT is the first systematic study of zero-shot tracking in MOT. It introduces a new path toward more generalizable and scalable tracking systems and offers tangible benefits for data construction. By using its zero-shot ability to generate high-quality tracking pre-annotations, manual labeling costs can be greatly reduced, accelerating large-scale dataset creation and promoting progress across the MOT community.

We conduct zero-shot evaluations on the DanceTrack, UAVDT, and BDD100K benchmarks. Without fine-tuning, SAM2MOT achieves state-of-the-art results, surpassing all closed-set fine-tuned methods and showing clear advantages in identity association. These results provide strong evidence for the value of segmentation in MOT and may encourage the community to revisit its role as a core tracking approach.

In conclusion, our contributions are as follows:

- **A Novel Segmentation-Tracking Perspective.** Breaking MOT’s stagnation within detection–association frameworks, we introduce a segmentation-driven tracking paradigm that offers new methodological insights and overcomes data dependence. Unlike previous segmentation-auxiliary approaches, we present the first segmentation-centric framework that systematically tackles major MOT challenges such as large-scale false positives and occlusions.
- **Zero-Shot Tracking Breakthrough.** We architecturally integrate a pre-trained detector and a pre-trained segmentor with tracking logic to achieve a zero-shot MOT system, eliminating scenario-specific fine-tuning with labeled data and marking a disruptive advancement.
- **SOTA Benchmark Results.** SAM2MOT achieves state-of-the-art performance on DanceTrack, UAVDT, and BDD100K. Notably, it reaches 75.8 HOTA and 83.9 IDF1 on DanceTrack, surpassing the second-best method by 2.1 and 4.5, respectively.

Related Works

Conventional MOT Paradigms

Tracking-by-Detection methods are exemplified by SORT(Bewley et al. 2016), which establishes the paradigm by associating detection boxes across frames using motion modeling and the Hungarian algorithm. Building on this, DeepSORT(Veeramani, Raymond, and Chanda 2018) integrates ReID features to enhance identity matching robustness. FairMOT(Zhang et al. 2021) is the first to unify detection and ReID in an end-to-end framework. ByteTrack(Zhang et al. 2022) improves robustness in

crowded scenes by leveraging low-confidence detections. OC-SORT(Cao et al. 2023) proposes an observation-centric Kalman update strategy to better handle occlusions, while StrongSORT(Du et al. 2023) introduces camera motion compensation to improve stability under dynamic viewpoints.

Although these methods have improved in accuracy and robustness, their tight coupling between detection and association makes performance highly dependent on detector quality, demanding extensive scene-specific tuning and hindering generalization.

In contrast, Tracking-by-Query methods use query-based mechanisms to detect new objects and maintain existing tracks within end-to-end transformer architectures. They fall into parallel and sequential categories: VisTR(Wang et al. 2021) and SeqFormer(Wu et al. 2022) (parallel) process short clips in one pass but suffer from high memory costs, while TransTrack(Sun et al. 2020), TrackFormer(Meinhardt et al. 2022), and MOTR(Zeng et al. 2022) (sequential) update queries frame by frame. To enhance temporal modeling, MeMOT(Cai et al. 2022) and MeMOTR(Gao and Wang 2023) add memory modules or long-term feature injection, and MOTRv3(Yu et al. 2023) resolves identity conflicts between new and existing objects through improved supervision.

Nevertheless, Tracking-by-Query methods remain limited in real-world scenarios due to their heavy reliance on large-scale, high-quality labeled tracking data, which is often costly and scarce.

Segmentation in MOT

Previous methods that incorporate segmentation mostly treat it an auxiliary information source, typically as a zero-shot alternative to ReID, to enhance object association modules. For example, MASA(Li et al. 2024) leverages appearance masks extracted by SAM(Kirillov et al. 2023) to improve the robustness of cross-frame identity matching.

Additionally, several studies explore segmentation-based tracking systems. TrackAnything(Yang et al. 2023) combines SAM with XMEM(Cheng and Schwing 2022) for interactive video object tracking and segmentation, while STAnything(Cheng et al. 2023) further integrates SAM, DeAOT(Yang and Yang 2022), and Grounding-DINO(Liu et al. 2023) to build a comprehensive zero-shot tracking framework.

However, we contend that these methods mostly stay at the level of system integration and functional assembly, lacking in-depth modeling or methodological exploration of segmentation as a core tracking mechanism. They often decompose multi-object tracking into independent single-object instances, failing to address challenges like occlusion and false positives, thereby making direct comparison with conventional paradigms difficult. Moreover, with the advent of SAM2(Ravi et al. 2024), which surpasses earlier combinations such as SAM+XMEM or DeAOT in both structure and performance, the relevance and research value of these approaches have further declined.

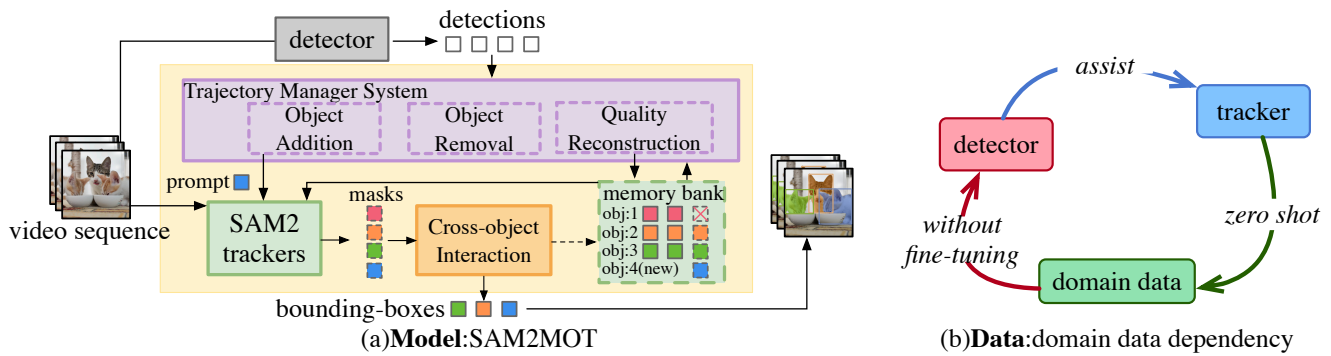


Figure 2: Overview of our *Tracking-by-Segmentation* MOT framework, including: (a) the overall architecture of SAM2MOT; (b) analysis of the relationship between detector, tracker, and data dependency.

SAM2 in Tracking

SAM2 (Ravi et al. 2024) extends segmentation from static images to video sequences through a memory bank mechanism, forming an interactive object tracking framework that achieves state-of-the-art performance in single-object video segmentation. Its pipeline starts with a user-provided prompt (points, boxes, or masks) to initialize tracking, extracting appearance features from the key frame and storing them in the memory bank as references. During inference, each frame references the key frame and the previous six frames, performing spatiotemporal matching to predict the object mask and confidence score. The new segmentation is then encoded and added to the memory bank, continually updating and refining the object’s appearance representation.

The memory frame selection strategy in SAM2 is overly simplistic. Subsequent works address this limitation: SAM2-Long(Ding et al. 2024) employs a decision-tree memory structure for long-range tracking, while SAMURAI(Yang et al. 2024a) improves performance through memory pruning that discards low-confidence entries. Together, these methods continually advance the state of the art in single-object tracking benchmarks.

Method

Overview

We propose SAM2MOT, focusing on two key challenges: (1) uncovering the role of segmentation in enhancing multi-object tracking, and (2) designing a complete tracking system with zero-shot generalization capability.

Empirically, we find that SAM2, with its pixel-level appearance modeling and implicit motion encoding, provides stronger robustness in single-object tracking than existing methods. However, naïvely extending MOT into multiple parallel single-object trackers introduces major limitations—particularly under occlusion, where degraded appearance quality and missing contextual cues frequently cause identity switches and tracking failures. To address this, we further introduce a Cross-object Interaction module (Section 3.2), which enables communication across objects during parallel tracking and jointly optimizes SAM2’s memory

frame selection strategy. This module effectively mitigates occlusion-induced errors and enhances overall tracking accuracy.

Meanwhile, SAM2 inherently operates as an interactive framework relying on user-provided prompts (e.g., boxes, points, or masks). Although a single prompt suffices for single-object tracking, multi-object tracking (MOT) involves the dynamic appearance and disappearance of objects, rendering static initialization inadequate. Therefore, we incorporate a pretrained detector to dynamically generate the required prompts for SAM2. Unlike traditional detection–association paradigms, SAM2 independently generates per-object tracking masks instead of directly inheriting bounding boxes from the detector. This design makes the system less dependent on the precision of a pretrained detector—it merely requires object recognition capability without accurate frame-by-frame localization. Consequently, the SAM2MOT framework achieves zero-shot generalization and decouples tracking performance from specific training data.

Nevertheless, directly connecting a pretrained detector with SAM2 introduces new challenges, including large-scale false positives and degraded long-term tracking quality. To mitigate these issues, we propose the Trajectory Manager System (Section 3.3), which provides a unified mechanism for managing object addition, removal, and quality reconstruction. The overall system architecture is illustrated in Figure 2, outlining the integration of the detector, SAM2, and the proposed modules into a unified zero-shot tracking framework.

Cross-object Interaction

Occlusion remains a fundamental challenge in object tracking. When an object B is occluded by another instance object A, SAM2 may erroneously associate the occluded object with the occluder, leading to identity confusion or mismatches. Such identity switches are particularly problematic, as they often propagate over time and prevent object B from being correctly recovered.

The root cause lies in SAM2’s memory bank strategy, which employs a fixed frame selection mechanism: the des-

ignated key frame and its six most recent reference frames are used to assist recognition in the current frame. Any error in these frames contaminates the memory bank, subsequently affecting future predictions and progressively compounding errors over time. In single-object tracking, a typical strategy is to discard low-confidence historical memories to limit error propagation. However, this fixed-threshold approach exhibits clear limitations in multi-object scenarios: a high threshold may prematurely remove valuable information, whereas a low threshold allows erroneous memories to persist. The problem becomes particularly severe during partial occlusions, where tracking errors are more frequent and fixed thresholds fail to distinguish between valid and corrupted memories.

Unlike single-object tracking, SAM2MOT can simultaneously track multiple objects within a frame. Leveraging this capability, we introduce a Cross-object Interaction Module, which enables information exchange between occluding and occluded instances. This mechanism dynamically identifies corrupted tracks during conflicts and removes compromised memory entries in real time, thereby enhancing both tracking accuracy and stability.

Specifically, to detect tracking failures where two objects (e.g., A and B) are mistakenly assigned to the same instance due to occlusion, we measure the overlap between their segmentation masks using the Mask Intersection over Union (mIoU). When the mIoU exceeds 0.8, it indicates substantial overlap, signaling that both trackers are recognizing the same object.

To further determine which object has been incorrectly identified, we design a two-stage analysis method. Assuming object B is misrecognized in the current frame, Stage 1 compares the segmentation confidence scores (i.e., the logit scores defined in SAM2) of objects A and B. If object A’s score is significantly higher (e.g., by more than two points), object B is directly flagged as the incorrectly tracked one. In Stage 2, when confidence scores are similar, the discrepancy usually arises because object A’s confidence declines gradually due to memory degradation, whereas object B’s confidence drops sharply due to sudden occlusion. Based on this observation, we compute the variance of each object’s confidence score over the past N frames (set to 10 in our implementation, see Equation 1). A steadily declining score in object A yields a lower variance, allowing object B to be identified as the erroneously tracked instance.

$$\sigma_{\text{logits}}^2 = \frac{1}{N} \sum_{i=1}^N (\text{logits}_i - \bar{\text{logits}})^2 \quad (1)$$

As shown in Figure 3, once object B is identified as having an identity confusion, its current-frame memory information is excluded from memory bank updates to prevent error propagation. Meanwhile, we maintain the filtering mechanism for low-confidence entries but adopt a relatively low threshold, which removes severely degraded appearance features while preserving temporal continuity as much as possible. By combining these two strategies, object B’s memory bank retains only accurate and reliable historical features,

enabling robust re-identification and stable tracking even after temporary recognition failures caused by occlusion.

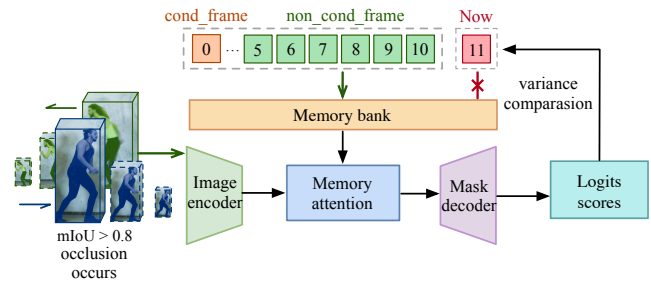


Figure 3: Cross-object interaction pipeline. During motion, when severe occlusion exceeds a predefined threshold(0.8), we identify identity-confused objects by analyzing their logits scores and corresponding variance. The memory entries of such objects in the current frame are then excluded from being written into the memory bank to prevent the propagation of incorrect information.

Trajectory Manager System

The motivation for the Trajectory Manager System is presented in Section 3.1. The following subsection details its mechanisms and implementation.

Object addition dynamically generates reliable initial prompts for newly appearing objects, for two main reasons: (1) Incorrect initialization prompts can cause severe and persistent false tracking in SAM2. Unlike detection–association paradigms, which rely on per-frame detection and can naturally discard false positives if not detected later, SAM2 primarily depends on the initial prompt—once incorrect, the resulting false object will continue to be tracked and is difficult to correct. (2) Low-quality appearance features (e.g., local features) used for tracking guidance easily cause identity confusion during occlusions or appearance changes.

To address this, SAM2MOT employs a three-stage filtering mechanism built upon the pre-trained detector, enforcing strict criteria to ensure that the generated prompts are both accurate and stable.

Specifically, in each frame, SAM2 generates segmentation masks M and corresponding tracking boxes for already tracked objects, while the detector outputs a set of detection boxes. The filtering process consists of three steps:

1. Discard low-confidence detection boxes using a confidence threshold (e.g., 0.5);
2. Match the remaining high-confidence boxes with SAM2’s tracking boxes using the Hungarian algorithm, treating unmatched boxes as candidate new objects;
3. Aggregate and invert all tracked-object masks M to form M_{non} (Equation 2), representing pixels not occupied by tracked objects. For each candidate, compute the overlap area p between its detection box and M_{non} . If the ratio $p/\text{area}(\text{box})$ exceeds a predefined threshold r (e.g., 0.7), the object is considered sufficiently distinct and initialized as new.

$$\mathcal{M}_{\text{non}} = I - \bigcup_{i=1}^n \mathcal{M}_i \quad (2)$$

Object removal is achieved by setting a frame-based tolerance threshold: if an object remains in a lost state beyond this limit, it is regarded as disappeared and removed. In detection–association paradigms, the lost state is easily determined—if the object is not detected, it is considered lost. However, in SAM2MOT, since SAM2 continuously outputs segmentation results, it may still produce low-confidence masks even when the object has disappeared. Therefore, the key challenge is to accurately determine whether the object is truly lost.

SAM2 outputs a confidence (logit) score for each mask. Based on the score distribution, we define three thresholds— τ_r , τ_p and τ_s —to categorize object states into four levels: reliable, pending, suspicious, and lost (Equation 3).

$$\text{State} = \begin{cases} \text{reliable,} & \text{logits} > \tau_r, \\ \text{pending,} & \tau_p < \text{logits} \leq \tau_r, \\ \text{suspicious,} & \tau_s < \text{logits} \leq \tau_p, \\ \text{lost,} & \text{logits} \leq \tau_s. \end{cases} \quad (3)$$

A reliable state indicates clear visibility and no occlusion; a pending state suggests mild occlusion or memory degradation, requiring appearance quality reconstruction (described later). A suspicious state usually arises from severe occlusion, where only partial appearance remains—this case is not explicitly handled in this work but remains open for future extensions. The lost state denotes that the object is almost completely occluded or has disappeared, rendering its appearance unavailable.

Quality reconstruction addresses the gradual degradation of prompt information caused by variations in object size, shape, appearance, and position. Since SAM2 references the key frame across subsequent frames, degraded prompts can directly impair tracking. While the original SAM2 allows manual updates of prompt information, SAM2MOT automates this process by leveraging redundant detection cues from the integrated detector.

To minimize computational overhead, reconstruction is triggered only when necessary under two conditions: (1) the object is in the “pending” state; and (2) the current tracking box successfully matches a high-confidence detection box (from the Object Addition stage), ensuring reliable updates. When both conditions are met, the matched high-confidence box updates the object’s keyframe information.

Experiments

Benchmarks

We evaluate SAM2MOT on three representative benchmark datasets: DanceTrack (Sun et al. 2022), UAVDT-MOT (Du et al. 2018), and BDD100K-MOT (Yu et al. 2020). These datasets collectively encompass a wide spectrum of tracking challenges, including occlusion, diverse motion patterns, and high inter-object similarity. Moreover, they span distinct

real-world scenarios—such as indoor dance performances, aerial drone perspectives, and autonomous driving environments—thereby providing a comprehensive evaluation of the model’s adaptability and robustness.

Evaluation Metrics. We adopt HOTA, MOTA, and IDF1 as the primary metrics to comprehensively evaluate the tracking performance of SAM2MOT. For DanceTrack, we further incorporate the sub-metrics DetA and AssA to assess detection and association performance at a finer granularity. Following the official evaluation protocols for BDD100K-MOT and UAVDT-MOT, we report only MOTA and IDF1 to ensure a consistent and fair comparison.

Implementation Details

We adopt Co-DINO-L(pretrained on COCO)(Zong, Song, and Liu 2023) and Grounding-DINO-L(pretrained on COCO and Object365)(Liu et al. 2023) as the object detectors, and directly applied to all benchmark datasets during testing without additional fine-tuning. Meanwhile, SAM2 utilizes the pretrained SAM2.1-large weights to ensure high-quality segmentation results.

Hyperparameter Settings. A unified configuration is applied across all datasets. In the Cross-object Interaction module, the window size for computing confidence variance is set to $N = 10$. In the Trajectory Manager System, for Object Addition, the high-confidence threshold is set by default to 0.5 for Co-DINO-L and 0.4 for Grounding-DINO-L, with an overlap ratio threshold of $r = 0.7$. For Object Removal, the tolerance is fixed at 25 frames, and object state classification is performed based on logit-based thresholds, with $\tau_r = 8.0$, $\tau_p = 6.0$, and $\tau_s = 2.0$.

State-of-the-Art Comparison

Table 1 reports the results of SAM2MOT across different benchmarks, compared with other popular methods.

DanceTrack. The results show that SAM2MOT substantially surpasses existing baselines in handling non-linear object motion, establishing a new state of the art. With Co-DINO-L as the detector, SAM2MOT achieves HOTA = 75.5, MOTA = 89.2, and IDF1 = 83.4. Using Grounding-DINO-L, the corresponding scores are 75.8, 88.5 and 83.9, respectively. Compared to previous best-performing methods, SAM2MOT demonstrates a clear advantage in object association, with HOTA +2.1, IDF1 +4.5, and AssA +6.3.

UAVDT-MOT. On the UAVDT-MOT dataset, using Co-DINO-L as the detector, SAM2MOT achieves MOTA = 55.6 and IDF1 = 74.4 under the IoU = 0.5 evaluation standard. Considering annotation noise in the UAVDT ground truth, we additionally perform an evaluation with a relaxed threshold of IoU = 0.4, where MOTA and IDF1 further improve to 66.1 and 79.4, respectively. Even under the stricter IoU = 0.5 setting, SAM2MOT attains the best results among all methods. The relaxed IoU = 0.4 evaluation better reflects its actual tracking performance, with both metrics showing consistent and significant improvement.

BDD100K-MOT. On the BDD100K-MOT dataset, using Co-DINO-L as the detector, SAM2MOT achieves MOTA = 57.5 and IDF1 = 70.8 under the eight-class configuration.

Methods	Publication	HOTA↑	MOTA↑	IDF1↑	AssA↑	DetA↑	FN↓	FP↓	IDSW ↓
<i>DanceTrack test set</i>									
SORT(Bewley et al. 2016)	ICIP2016	47.9	91.8	50.8	31.2	72.0	-	-	-
DeepSORT(Veeramani, Raymond, and Chanda 2018)	ICIP2017	45.6	87.8	47.9	29.7	71.0	-	-	-
FairMOT(Zhang et al. 2021)	IJCV2021	39.7	82.2	40.8	23.8	66.7	-	-	-
CenterTrack(Zhou, Koltun, and Krähenbühl 2020)	ECCV2020	41.8	86.8	35.7	22.6	78.1	-	-	-
GTR(Zhou et al. 2022)	CVPR2022	48.0	84.7	50.3	31.9	72.5	-	-	-
ByteTrack(Zhang et al. 2022)	ECCV2022	47.3	89.5	52.5	31.4	71.6	-	-	-
MOTR(Zeng et al. 2022)	ECCV2022	54.2	79.7	51.5	40.2	73.5	-	-	-
MOTRv2(Zhang, Wang, and Zhang 2023)	CVPR2022	69.9	91.9	71.7	59.0	83.0	-	-	-
CoTrack(Liu, Wu, and Fu 2023)	ICCV2023	72.6	92.1	74.0	62.3	-	-	-	-
FineTrack(Ren et al. 2023)	CVPR2023	52.7	89.9	59.8	38.5	72.4	-	-	-
OC-SORT(Cao et al. 2023)	CVPR2023	54.6	89.6	54.6	40.2	80.4	-	-	-
DiffMOT(Lv et al. 2024)	CVPR2024	62.3	92.8	63.0	47.2	82.5	-	-	-
Hybrid-SORT(Yang et al. 2024b)	AAAI2024	65.7	91.8	67.4	-	-	-	-	-
AED(Fang et al. 2025)	TIP2025	66.6	92.2	69.7	54.3	82.0	-	-	-
MOTIP(Gao, Qi, and Wang 2025)	CVPR2025	73.7	92.7	79.4	65.9	82.6	-	-	-
SAM2MOT(<i>Co-DINO-L</i> , <i>no fine-tuning</i>)	Ours	75.5	89.2	83.4	71.3	80.3	-	-	-
SAM2MOT(<i>Grounding-DINO-L</i> , <i>no fine-tuning</i>)	Ours	75.8	88.5	83.9	72.2	79.7	-	-	-
<i>UAVDT-MOT test set</i>									
OC-SORT(Cao et al. 2023)	CVPR2023	-	47.5	64.9	-	-	148378	47681	288
UAVMOT(Liu et al. 2022)	CVPR2022	-	46.4	67.3	-	-	115940	66352	456
FOLT(Yao et al. 2023)	MM2023	-	48.5	68.3	-	-	155696	36429	338
GLOA(Shi et al. 2023)	J-STARs2023	-	49.6	68.9	-	-	115567	55822	433
DroneMOT(Wang, Wang, and Li 2024)	ICRA2024	-	50.1	69.6	-	-	112548	57411	129
SAM2MOT(<i>iou0.5</i> , <i>Co-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	55.6	74.4	-	-	92504	58610	141
SAM2MOT(<i>iou0.5</i> , <i>Grounding-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	51.0	71.7	-	-	103977	62906	139
SAM2MOT(<i>iou0.4</i> , <i>Co-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	66.1	79.3	-	-	74586	40692	136
SAM2MOT(<i>iou0.4</i> , <i>Grounding-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	60.9	76.6	-	-	87003	45932	155
<i>BDD100K-MOT val set</i>									
QDtrack(<i>cls8</i>) (Fischer et al. 2023)	TPAMI2023	-	63.5	71.5	-	-	-	-	6262
Unicorn(<i>cls8</i>) (Yan et al. 2022)	ECCV2022	-	66.6	71.3	-	-	-	-	10876
MOTRv2(<i>cls8</i>) (Zhang, Wang, and Zhang 2023)	CVPR2022	-	65.6	72.7	-	-	-	-	-
UNINEXT(<i>cls8</i>) (Yan et al. 2023)	CVPR2023	-	67.1	69.9	-	-	-	-	10222
MASA(<i>cls8</i>) (Li et al. 2024)	CVPR2024	-	-	71.7	-	-	-	-	-
SAM2MOT(<i>cls-8</i> , <i>Co-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	57.5	70.8	-	-	-	-	5587
SAM2MOT(<i>cls8</i> , <i>Grounding-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	44.1	63.6	-	-	-	-	5184
SAM2MOT(<i>cls-3</i> , <i>Co-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	63.0	73.7	-	-	-	-	5755
SAM2MOT(<i>cls3</i> , <i>Grounding-DINO-L</i> , <i>no fine-tuning</i>)	Ours	-	58.0	71.0	-	-	-	-	5464

Table 1: Comparison with other popular MOT methods on different benchmarks.

When evaluated under the three-class setting, the results improve to MOTA = 63.0 and IDF1 = 73.7. The relatively lower performance in the 8-class setting is attributed to the absence of fine-tuning, which leads the pre-trained detector to confuse visually similar categories. In contrast, under the simplified 3-class scheme, SAM2MOT maintains strong IDF1 performance and significantly reduces ID switches, demonstrating its robust identity association capability.

Ablation Study and Analysis

Unified Detector Comparison. To assess SAM2MOT’s sensitivity to detection accuracy, we compare its performance with two representative detection–association methods—ByteTrack and OC-SORT—using the same detector configuration. Other end-to-end approaches are excluded from this comparison since they do not depend on an independent detector. The experimental setup follows the same configuration described in Section 4.3.

As shown in Table 2, under the same detector and unified settings, SAM2MOT consistently outperforms ByteTrack and OC-SORT across all benchmarks, exhibiting a particularly strong advantage in identity association metrics.

Moreover, the results indicate that, compared with con-

ventional detection–association methods, SAM2MOT can recover additional true positives (TPs) through its internally generated bounding boxes, even when the detector fails—effectively reducing its reliance on precise detection outputs.

Component Ablation. We evaluate the contribution of each module in SAM2MOT on the DanceTrack test set, and the results are summarized in Table 3. The evaluation focuses on three core components: Object Addition (Add), Cross-object Interaction (CoI), and Quality Reconstruction (Q-R). Among them, the Cross-object Interaction module provides the most significant contribution to association accuracy, enabling SAM2MOT to achieve a breakthrough improvement in object correlation.

Limitation.

Inference Efficiency.

The primary limitation of the current architecture lies in its inference speed. We argue that strategically trading off some runtime efficiency to validate the framework’s potential—similar to pioneering efforts—is both reasonable and worthwhile. Nevertheless, we are actively investigating ac-

Methods	HOTA \uparrow	MOTA \uparrow	IDF1 \uparrow	AssA \uparrow	DetA \uparrow	TP \uparrow	FN \downarrow	FP \downarrow	IDSW \downarrow	MT \uparrow	ML \downarrow
<i>DanceTrack test set</i>											
ByteTrack(<i>Co-DINO-L</i>)	56.1	87.2	56.6	40.3	78.2	259793	29373	6028	1650	240	7
ByteTrack(<i>Grounding-DINO-L</i>)	53.3	86.8	53.7	37.0	77.0	260160	29006	7281	1864	233	5
OC-SORT(<i>Co-DINO-L</i>)	56.2	86.5	56.8	40.7	77.8	-	-	-	-	-	-
OC-SORT(<i>Grounding-DINO-L</i>)	53.6	84.4	56.6	38.8	74.3	-	-	-	-	-	-
SAM2MOT(<i>Co-DINO-L</i>)	75.5	89.2	83.4	71.3	80.3	274582	14584	15653	854	273	2
SAM2MOT(<i>Grounding-DINO-L</i>)	75.8	88.5	83.9	72.2	79.7	271472	17694	14650	879	264	3
<i>UAVDT-MOT test set</i>											
ByteTrack(<i>iou0.5, Co-DINO-L</i>)	-	52.4	67.8	-	-	204320	136586	25081	452	588	250
ByteTrack(<i>iou0.5, Grounding-DINO-L</i>)	-	44.3	61.6	-	-	172820	168086	21533	414	465	318
ByteTrack(<i>iou0.4, Co-DINO-L</i>)	-	55.4	69.1	-	-	209303	131603	20098	482	607	240
ByteTrack(<i>iou0.4, Grounding-DINO-L</i>)	-	46.4	62.8	-	-	176507	164399	17846	477	488	306
OC-SORT(<i>iou0.5, Co-DINO-L</i>)	-	48.5	64.5	-	-	189117	151789	23416	444	534	276
OC-SORT(<i>iou0.5, Grounding-DINO-L</i>)	-	39.5	56.9	-	-	154404	186502	19348	368	413	363
OC-SORT(<i>iou0.4, Co-DINO-L</i>)	-	50.8	65.6	-	-	193014	147892	19519	482	555	263
OC-SORT(<i>iou0.4, Grounding-DINO-L</i>)	-	41.1	57.8	-	-	157100	183806	16652	417	425	355
SAM2MOT(<i>iou0.5, Co-DINO-L</i>)	-	55.6	74.4	-	-	248402	92504	58610	141	742	161
SAM2MOT(<i>iou0.5, Grounding-DINO-L</i>)	-	51.0	71.7	-	-	236929	103977	62906	139	694	189
SAM2MOT(<i>iou0.4, Co-DINO-L</i>)	-	66.1	79.3	-	-	266320	74586	40692	136	816	147
SAM2MOT(<i>iou0.4, Grounding-DINO-L</i>)	-	60.9	76.6	-	-	253903	87003	45932	155	767	171
<i>BDD100K-MOT val set</i>											
ByteTrack(<i>cls-3, Co-DINO-L</i>)	-	54.7	60.7	-	-	-	149852	25710	24832	-	-
ByteTrack(<i>cls-3, Grounding-DINO-L</i>)	-	44.8	55.6	-	-	-	179103	43437	21680	-	-
SAM2MOT(<i>cls-3, Co-DINO-L</i>)	-	63.0	73.7	-	-	-	125064	32965	5755	-	-
SAM2MOT(<i>cls-3, Grounding-DINO-L</i>)	-	58.0	71.0	-	-	-	138491	42070	5464	-	-

Table 2: Comparison with ByteTrack and OC-SORT using the same detector on different benchmarks.

Baseline	Add	CoI	Q-R	HOTA	MOTA	IDF1
<i>Co-DINO-L</i>						
✓				62.9	55.6	69.6
✓	✓			67.9	69.7	74.4
✓	✓		✓	69.1	69.2	76.0
✓	✓	✓		73.8	87.4	80.9
✓	✓	✓	✓	75.5	89.2	83.4
<i>Grounding-DINO-L</i>						
✓				60.9	50.1	66.5
✓	✓			67.4	67.2	74.0
✓	✓		✓	69.0	68.7	76.2
✓	✓	✓		73.6	86.5	81.2
✓	✓	✓	✓	75.8	88.5	83.9

Table 3: Ablation study on different strategies, and evaluation metrics on DanceTrack test set.

celeration strategies along three directions:

1. Replacing the detection backbone with lighter-weight variants (e.g., YOLOE (Wang et al. 2025));
2. Accelerating the segmentation module by leveraging techniques such as EdgeTAM (Zhou et al. 2025), which enhances the efficiency of SAM2-based architectures;
3. Minimizing system-level latency through parallel inference across multiple SAM2 instances.

Furthermore, despite its limited inference efficiency, SAM2MOT demonstrates strong zero-shot generalization capability, effectively mitigating the key bottleneck of video tracking data pre-annotation. This substantially reduces manual labeling effort and provides a practical pathway for constructing large-scale tracking datasets within the community.

Holistic Object Modeling.

SAM2MOT lacks holistic object modeling. When only part of an object is visible (e.g., the head while the body is occluded), it tracks stably but the box covers only the visible region. Detection–association methods, by contrast, infer full bounding boxes as their detectors capture the object’s entire extent. This reflects a core limitation of segmentation-driven frameworks, especially evident on benchmarks like MOTChallenge.

Long-term Memory.

SAM2MOT also struggles with long-term memory. Extended occlusions often cause tracking loss since SAM2, built on short-term memory, encodes only adjacent-frame motion without modeling long-term dependencies. Although SAM2 retains limited long-term capacity, it is unstable. We argue that a more adaptive memory-frame selection could leverage appearance cues for implicit re-identification, improving robustness in long-duration tracking.

Conclusion

We propose SAM2MOT, a segmentation-driven paradigm for multi-object tracking that encourages the community to revisit the role of segmentation for tracking. Furthermore, we design a unified tracking framework with strong zero-shot generalization capability, effectively addressing performance degradation under data-scarce conditions and enhancing practical applicability. Extensive experiments on multiple mainstream benchmarks validate the superior identity association capability of SAM2MOT.

References

- Bergmann, P.; et al. 2019. Tracking without bells and whistles. In *Proceedings of the IEEE/CVF international conference on computer vision*, 941–951.
- Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; and Upcroft, B. 2016. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, 3464–3468. Ieee.
- Cai, J.; Xu, M.; Li, W.; Xiong, Y.; Xia, W.; Tu, Z.; and Soatto, S. 2022. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8090–8100.
- Cao, J.; Pang, J.; Weng, X.; Khirodkar, R.; and Kitani, K. 2023. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9686–9696.
- Cheng, H. K.; and Schwing, A. G. 2022. XMemo: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model. In *ECCV*.
- Cheng, Y.; Li, L.; Xu, Y.; Li, X.; Yang, Z.; Wang, W.; and Yang, Y. 2023. Segment and Track Anything. *arXiv preprint arXiv:2305.06558*.
- Ding, S.; Qian, R.; Dong, X.; Zhang, P.; Zang, Y.; Cao, Y.; Guo, Y.; Lin, D.; and Wang, J. 2024. Sam2long: Enhancing sam 2 for long video segmentation with a training-free memory tree. *arXiv preprint arXiv:2410.16268*.
- Du, D.; Qi, Y.; Yu, H.; Yang, Y.; Duan, K.; Li, G.; Zhang, W.; Huang, Q.; and Tian, Q. 2018. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, 370–386.
- Du, Y.; Zhao, Z.; Song, Y.; Zhao, Y.; Su, F.; Gong, T.; and Meng, H. 2023. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 25: 8725–8737.
- Fang, Z.; Liang, C.; Zhou, X.; Zhu, S.; and Li, X. 2025. Associate everything detected: Facilitating tracking-by-detection to the unknown. *IEEE Transactions on Image Processing*.
- Fischer, T.; Huang, T. E.; Pang, J.; Qiu, L.; Chen, H.; Darrell, T.; and Yu, F. 2023. Qdtrack: Quasi-dense similarity learning for appearance-only multiple object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12): 15380–15393.
- Gao, R.; Qi, J.; and Wang, L. 2025. Multiple object tracking as id prediction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 27883–27893.
- Gao, R.; and Wang, L. 2023. MeMOTR: Long-term memory-augmented transformer for multi-object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9901–9910.
- Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4015–4026.
- Li, S.; Ke, L.; Danelljan, M.; Piccinelli, L.; Segu, M.; Van Gool, L.; and Yu, F. 2024. Matching anything by segmenting anything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18963–18973.
- Liu, S.; Li, X.; Lu, H.; and He, Y. 2022. Multi-object tracking meets moving UAV. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8876–8885.
- Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Li, C.; Yang, J.; Su, H.; Zhu, J.; et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.
- Liu, Y.; Wu, J.; and Fu, Y. 2023. Collaborative tracking learning for frame-rate-insensitive multi-object tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9964–9973.
- Lv, W.; Huang, Y.; Zhang, N.; Lin, R.-S.; Han, M.; and Zeng, D. 2024. Diffmot: A real-time diffusion-based multiple object tracker with non-linear prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19321–19330.
- Meinhardt, T.; Kirillov, A.; Leal-Taixe, L.; and Feichtenhofer, C. 2022. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8844–8854.
- Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; et al. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*.
- Ren, H.; Han, S.; Ding, H.; Zhang, Z.; Wang, H.; and Wang, F. 2023. Focus on details: Online multi-object tracking with diverse fine-grained representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11289–11298.
- Shi, L.; Zhang, Q.; Pan, B.; Zhang, J.; and Su, Y. 2023. Global-local and occlusion awareness network for object tracking in UAVs. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16: 8834–8844.
- Sun, P.; Cao, J.; Jiang, Y.; Yuan, Z.; Bai, S.; Kitani, K.; and Luo, P. 2022. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 20993–21002.
- Sun, P.; Cao, J.; Jiang, Y.; Zhang, R.; Xie, E.; Yuan, Z.; Wang, C.; and Luo, P. 2020. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*.
- Veeramani, B.; Raymond, J. W.; and Chanda, P. 2018. DeepSort: deep convolutional networks for sorting haploid maize seeds. *BMC bioinformatics*, 19: 1–9.
- Wang, A.; Liu, L.; Chen, H.; Lin, Z.; Han, J.; and Ding, G. 2025. Yoloe: Real-time seeing anything. *arXiv preprint arXiv:2503.07465*.
- Wang, P.; Wang, Y.; and Li, D. 2024. Dronemot: Drone-based multi-object tracking considering detection difficulties and simultaneous moving of drones and objects. In *2024*

- IEEE International Conference on Robotics and Automation (ICRA)*, 7397–7404. IEEE.
- Wang, Y.; Xu, Z.; Wang, X.; Shen, C.; Cheng, B.; Shen, H.; and Xia, H. 2021. End-to-end video instance segmentation with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8741–8750.
- Wu, J.; Cao, J.; Song, L.; Wang, Y.; Yang, M.; and Yuan, J. 2021. Track to detect and segment: An online multi-object tracker. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12352–12361.
- Wu, J.; Jiang, Y.; Bai, S.; Zhang, W.; and Bai, X. 2022. Seqformer: Sequential transformer for video instance segmentation. In *European Conference on Computer Vision*, 553–569. Springer.
- Yan, B.; Jiang, Y.; Sun, P.; Wang, D.; Yuan, Z.; Luo, P.; and Lu, H. 2022. Towards grand unification of object tracking. In *European conference on computer vision*, 733–751. Springer.
- Yan, B.; Jiang, Y.; Wu, J.; Wang, D.; Luo, P.; Yuan, Z.; and Lu, H. 2023. Universal instance perception as object discovery and retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15325–15336.
- Yang, C.-Y.; Huang, H.-W.; Chai, W.; Jiang, Z.; and Hwang, J.-N. 2024a. Samurai: Adapting segment anything model for zero-shot visual tracking with motion-aware memory. *arXiv preprint arXiv:2411.11922*.
- Yang, J.; Gao, M.; Li, Z.; Gao, S.; Wang, F.; and Zheng, F. 2023. Track Anything: Segment Anything Meets Videos. *arXiv:2304.11968*.
- Yang, M.; Han, G.; Yan, B.; Zhang, W.; Qi, J.; Lu, H.; and Wang, D. 2024b. Hybrid-sort: Weak cues matter for online multi-object tracking. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 6504–6512.
- Yang, Z.; and Yang, Y. 2022. Decoupling Features in Hierarchical Propagation for Video Object Segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Yao, M.; Wang, J.; Peng, J.; Chi, M.; and Liu, C. 2023. Folt: Fast multiple object tracking from uav-captured videos based on optical flow. In *Proceedings of the 31st ACM International Conference on Multimedia*, 3375–3383.
- Yu, E.; Wang, T.; Li, Z.; Zhang, Y.; Zhang, X.; and Tao, W. 2023. Motrv3: Release-fetch supervision for end-to-end multi-object tracking. *arXiv preprint arXiv:2305.14298*.
- Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; and Darrell, T. 2020. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2636–2645.
- Zeng, F.; Dong, B.; Zhang, Y.; Wang, T.; Zhang, X.; and Wei, Y. 2022. Motr: End-to-end multiple-object tracking with transformer. In *European conference on computer vision*, 659–675. Springer.
- Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; and Wang, X. 2022. Bytetrack: Multi-object tracking by associating every detection box. In *European conference on computer vision*, 1–21. Springer.
- Zhang, Y.; Wang, C.; Wang, X.; Zeng, W.; and Liu, W. 2021. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International journal of computer vision*, 129: 3069–3087.
- Zhang, Y.; Wang, T.; and Zhang, X. 2023. Motrv2: Bootstrapping end-to-end multi-object tracking by pretrained object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 22056–22065.
- Zhou, C.; Zhu, C.; Xiong, Y.; Suri, S.; Xiao, F.; Wu, L.; Krishnamoorthi, R.; Dai, B.; Loy, C. C.; Chandra, V.; and Soran, B. 2025. EdgeTAM: On-Device Track Anything Model. *arXiv preprint arXiv:2501.07256*.
- Zhou, X.; Koltun, V.; and Krähenbühl, P. 2020. Tracking objects as points. In *European conference on computer vision*, 474–490. Springer.
- Zhou, X.; Yin, T.; Koltun, V.; and Krähenbühl, P. 2022. Global tracking transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8771–8780.
- Zong, Z.; Song, G.; and Liu, Y. 2023. Detsr with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF international conference on computer vision*, 6748–6758.